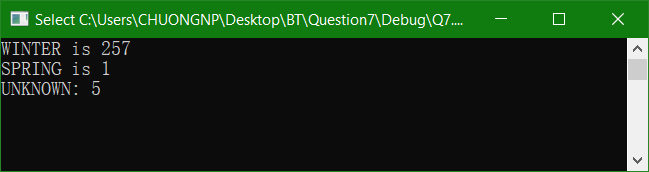
# Question 7: Enum & Union

**Q7.1: Explain the result**

|  |
| --- |
| #include <iostream>  using namespace std;  enum SEASON {SPRING = 0x1, SUMMER = 0x10, FALL = 0x100, WINTER};  void CheckSeason(int season)  {  switch (season)  {  case SPRING:  printf("SPRING is %d\n", SPRING);  break;  case SUMMER:  printf("SUMMER is %d\n", SUMMER);  break;  case FALL:  printf("FALL is %d\n", FALL);  break;  case WINTER:  printf("WINTER is %d\n", WINTER);  break;  default:  printf("UNKNOWN: %d", season);  break;  }  }  void main()  {  SEASON season = WINTER;  CheckSeason(season);  CheckSeason(1);  CheckSeason(5);  } |

**Kq:**



Kiểu Enum là kiểu giúp ta định nghĩa một tập hợp các hằng số nguyên.

* WINTER là một phần tử của enum SEASION, nó ở liền sau phần tử FALL (được khởi tạo bằng 0x100 = 256) nên nó có giá trị kế tiếp là 257.
* SPRING được khởi tạo = 1.
* 5 không được khởi tạo trong enum SEASION nên trong câu lệnh switch, trường hợp default được thực hiện.

**Q7.2: Alignment is divided into 2 group: vertical alignment and horizontal alignment. For horizontal alignment, we have values: LEFT, CENTER, and RIGHT. For vertical alignment, we have: TOP, CENTER, and BOTTOM.**

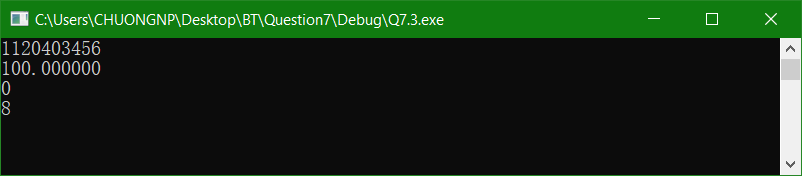
* **Try to define two enum Vertical\_Align and Horizontal\_Align with their above values**
* **Detect and solve problems when define enums.**

????????????????????

**Q7.3: Draw diagrams to explain the results:**

|  |
| --- |
| union MyUnion  {  int iVal;  float fVal;  char cVal[5];  };  void main()  {  union MyUnion u;  u.fVal = 100;  printf("%d\n%f\n%d\n%d\n", u.iVal, u.fVal, u.cVal[0], sizeof(u));  } |

**Kq:**



fVal

Memory block

iVal

|  |  |  |  |
| --- | --- | --- | --- |
| **42C80000** | | | |
| **42C80000** | | | |
| **42C80000** | | | |
| **42** | **C8** | **00** | **00** | ? |  |

cVal[5]